IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application No.: 10/821,412	§	Examiner:	Eland, Shawn
Filed: April 9, 2004	§	Group/Art Unit:	2188
Inventor(s):	§	Atty. Dkt. No:	5681-01601
Landin, et al.	§	Confirm No.	8427
	§		
	§		
	§		
Title: MULTI-NODE COMPUTER SYSTEM EMPLOYING A REPORTING MECHANISM FOR MULTI-NODE TRANSACTIONS	§		

APPEAL BRIEF

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir/Madam:

Further to the Notice of Appeal of March 26, 2008, Appellant presents this Appeal Brief. Appellant respectfully requests that this appeal be considered by the Board of Patent Appeals and Interferences.

I. REAL PARTY IN INTEREST

The subject application is owned by Sun Microsystems Inc.. An assignment of the present application to the owner is recorded at Reel 015764, Frame 0787.

II. RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences known to Appellant.

III. STATUS OF CLAIMS

Claims 1-38 are pending. Claims 1-38 are rejected under 35 U.S.C. § 103(a). It is these rejections that are being appealed. A copy of claims 1-38 is included in the Claims Appendix attached hereto.

IV. STATUS OF AMENDMEMNTS

No unentered amendment to the claims has been filed after final rejection. The Appendix hereto reflects the current state of the rejected claims.

V. SUMMARY OF THE CLAIMED SUBJECT MATTER

Independent claim 1 is directed to a system (*See* e.g., FIG. 20, #100) including a node (*See* e.g., FIG. 20, #140A) and an additional node (*See* e.g., FIG. 20, #140B) coupled by an inter-node network (*See* e.g., FIG. 20, #154). The node includes an active device (*See* e.g., FIG. 20, #142A, 146A), an interface (*See* e.g., FIG. 20, #148A) to the inter-node network, a system memory (*See* e.g., FIG. 20, #144A), and an address network (*See* e.g., FIG. 20, #150A) and a data network (*See* e.g., FIG. 20, #152A) that is separate from the address network (*See* e.g., specification page 11, lines 18-20), coupling the active device, the interface, and the system memory. The active device sends an address packet to initiate a transaction to gain an access right to a coherency unit (*See* e.g., specification page 70, lines 13-17). In response to receiving the address packet, the system memory is configured to send a report corresponding to the address packet to the interface if the transaction cannot be satisfied within the node (*See* e.g., specification page 70, lines 10-21). The interface is configured to ignore the address packet and to send a coherency message requesting the access right to the additional node via the inter-node network in response to the report (*See* e.g., specification page 70, lines 10-21; page 71, lines 9-14).

Independent claim 14 is directed to a node (*See* e.g., FIG. 20, #140A) for use in a multi-node system (*See* e.g., FIG. 20, #100). The node includes a plurality of devices including a system memory (*See* e.g., FIG. 20, #144A), an active device (*See* e.g., FIG. 20, #142A, 146A), and an interface (*See* e.g., FIG. 20, #148A) configured to send and receive coherency messages on an inter-node network (*See* e.g., FIG. 20, #154; specification page 63, lines 7-15) coupling nodes in the multi-node computer system. The node also includes an address network (*See* e.g., FIG. 20, #150A) configured to convey address packets between the plurality of devices (*See* e.g., specification page 11, lines 4-20). The node further includes a data network (*See* e.g., FIG. 20, #152A) that is separate from the address network and configured to convey data packets between the plurality of devices (*See* e.g., specification page 11, lines 4-20). In response to receiving from the active device an address packet initiating a transaction to gain an access right to

a coherency unit, the system memory is configured to send a report corresponding to the address packet to the interface if the transaction cannot be satisfied within the node (*See* e.g., specification page 70, lines 10-21). The interface is configured to ignore the address packet and to send a coherency message requesting the access right on the inter-node network in response to the report (*See* e.g., specification page 70, lines 10-21; page 71, lines 9-14).

Independent claim 26 is directed to a method of operating a multi-node system (*See* e.g., FIG. 20, #100), wherein the multi-node system includes a node (*See* e.g., FIG. 20, #140A) and an additional node (*See* e.g., FIG. 20, #140B) coupled by an inter-node network (*See* e.g., FIG. 20, #154). The method includes an active device (*See* e.g., FIG. 20, #142A, 146A) in the node initiating a transaction to gain an access right to a coherency unit by sending an address packet on an address network within the node (*See* e.g., specification page 70, lines 13-17). The method also includes an interface in the node ignoring the address packet (*See* e.g., specification page 70, lines 10-21). In response to said sending the address packet, a system memory in the node sending, via a data network that is separate from the address network using a data packet, a report corresponding to the address packet to the interface if the transaction cannot be satisfied within the node (*See* e.g., specification page 70, lines 14-18). In addition, in response to the report, the interface sending a coherency message to an additional interface in the additional node via the inter-node network, wherein the coherency message requests the access right to the coherency unit (*See* e.g., specification page 70, lines 20-21; page 71, lines 9-14).

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

1. Claims 1-38 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Liencres et al. (U.S. Patent No. 5,434,993) in view of Chandrasekaran et al. (U.S. Patent No. 6,970,872), and in further view of Roy (6,065,092).

VII. ARGUMENT

First Ground of Rejection:

Claims 1-38 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Liencres et al. (U.S. Patent No. 5,434,993) in view of Chandrasekaran et al. (U.S. Patent No. 6,970,872), and in further view of Roy (6,065,092). Appellant traverses this rejection for at least the following reasons.

Independent claims (by number):

Appellant respectfully submits that each of claims 1, 14, and 26 recites a combination of features not taught or suggested in Liencres, Chandrasekaran, or Roy. For example, claim 1 recites a combination of features including:

> a node including an active device, an interface to an inter-node network, a system memory, and an address network and a data network that is separate from the address network, coupling the active device, the interface, and the memory;
>
> ...
>
> wherein in response to receiving **from the active device** an address packet initiating a transaction to gain an access right to a coherency unit, the **system memory** is configured to **send a report** corresponding to the address packet to the interface **if the transaction cannot be satisfied within the node**;
>
> wherein the interface is configured to ignore the address packet and to send a coherency message requesting the access right to the additional node via the inter-node network in response to the report. (Emphasis added).

The Examiner asserts the combination of Liencres, Chandrasekaran and Roy teaches all the limitations recited in Applicant's claim 1. Applicant respectfully disagrees with at least portions of the Examiner's assertions.

More particularly, the Examiner asserts in the Advisory action dated March 18, 2008, and in the final Office action dated December 26, 2007 Liencres teaches Appellant's claimed address network and a separate data network that couples the active devices, system memory, and the interface. More particularly, the Examiner asserts that bus 33 of Liencres is Appellant's address network. Appellant disagrees.

FIG. 3a and 3b of Liencres clearly show that the memory alluded to by the Examiner is a cache memory 37, and not a system memory. In Appellant's claims, specification and drawings it is clear the claimed memory is a system memory and not a cache memory. Appellant submits Liencres clearly shows a system (main) memory coupled by a memory bus 25 to each node. Notwithstanding, the Examiner still insists that the cache memory 37 is analogous to Appellant's claimed memory. However, even if, *arguendo*, one were to suppose that the cache memory 37 were analogous to the Appellant's claimed system memory, the topology of Liencres is different. Specifically, the cache memory 37 is not coupled to the bus 33 but is instead coupled to the cache controller 35 and to processor 21 by a separate bus. As such, the bus cache controller 31, the processor 21 and the cache memory 37 are NOT all coupled together by bus 33 as suggested by the Examiner and required by Appellant's claims. Thus, Appellant fails to see how the system of Liencres teaches the structure recited in Appellant's claims. Appellant submits it does not.

In addition, the Examiner acknowledges Liencres does not teach "wherein in response to receiving from the active device an address packet initiating a transaction to gain an access right to a coherency unit, the system memory is configured to send a report corresponding to the address packet to the interface if the transaction cannot be satisfied within the node."

The Examiner also asserts Liencres teaches ignoring the address packet, but acknowledges Liencres does not teach in response to the report, the interface sending "a coherency message requesting the access right to the additional node via the inter-node network," as recited in claim 1.

However, the Examiner asserts Chandrasekaran teaches the above limitations at col. 6 lines 25-36. The Examiner asserts "When another node writes out data, it sends out a report stating the latest write time for that data. The read data is invalid once its timestamp comes before the latest write time. The node, now having an invalid read

data, will ignore the current address packet (col. 2, lines 60-66), and then have to request the updated data from the additional node. It would have been obvious… to employ optimistic reading of data using "write-time" validity checking so that reads could be employed when another node has exclusive access but hasn't yet written the data." Applicant respectfully disagrees with the Examiner's application of the Chandrasekaran art to Applicant's claims.

More particularly, Chandrasekaran is directed to optimistic reads and write time validity checking. Chandrasekaran discloses

> In an embodiment using the first type of validity checking, the time that the optimistic read is started is compared to the latest time that the data block was written by any of the other nodes. If the read was started after the last write, the read is valid. This can be determined even before the read is finished, but involves the writing node publishing its write time to the other nodes. A node can publish its write time in any way, such as by broadcasting the write time **to the other nodes**, by storing the write time and responding to requests **from other nodes**, or by **sending the write time to a lock manager**. This type of validity checking is called "write-time" validity checking herein. (See col. 6, lines 25-36) (Emphasis added)
>
> In step 210 a request for a lock to access a particular data block is sent. In embodiments that use a lock manager, the request is sent to the lock manager. In other embodiments, the lock manager may be omitted and the I/O processes **114** can communicate with each other to grant locks. In still other embodiments, other methods can be used to assure consistent content of a resource; and step 210 corresponds to initiating an operation to grant permission to access a resource in order to assure consistency of content of the resource. In some embodiments, a retrieval start time, indicating a time that step 220 is performed, is included in a message sent to a lock manager or the other nodes.
> In step 220, retrieval of the particular data block is started. For example, a call is made to an operating system function to move data from the disc data blocks corresponding to the particular data block into a location in the memory of one of the nodes. In some embodiments the location in memory is in the cache **112** for the I/O process **114** on the node; in some embodiments the location is a buffer outside the cache **112**. (See col. 7, lines 27-45) (Emphasis added)

> If it is determined in step 250 that the particular block returned in step 240 is not valid, then control passes to step 260. In step 260, the operation to retrieve the particular block from disk or a remote cache is started again, based on permission received in step 230. Because permission has been received in step 230 before performing step 260, the data block received in response to step 260 will be valid. In embodiments in which permission is not received in step 230, such as embodiments in which a message denying permission to access the particular data block is received in step 230, step 260 is delayed until permission is received. (See col. 8 lines 56-66) (Emphasis added)

From the foregoing description, Applicant submits Chandrasekaran is disclosing a given node broadcasting or somehow publishing its write time for a write of a given data block. That write time may be stored and then compared or it may be sent in response to a request for access. Some other node performs an optimistic read of the data, and if that read occurs earlier in time than the write, the read data would be invalid. The write time may be kept by **a lock manager** which returns the write validity information. Furthermore, Chandrasekaran is disclosing in response to a request for the data block, the data retrieval process is started irrespective of whether the data is valid (i.e., optimistically). Accordingly, the only way the data retrieval process can be started is if the data block is available. Now the data may not be valid due to the write time constraints, but it is still available, and the memory system does not have knowledge of that validity. This is in contrast to an active device within a the node requesting an access right to a coherency unit, and the system memory in that node responding by sending "a report," **which is a special type of data packet (and it is not the requested data),** to an interface, "if the transaction cannot be satisfied within the node." In addition, the interface ignores the address packet (since it came from within the node), but responds to the report by sending a coherency message out to the other nodes. Neither of these limitations is taught or suggested by Chandrasekaran.

Furthermore, since the Examiner is asserting that cache memory 37 of Liencres is the system memory recited in Appellant's claims, Appellant submits, Chandrasekaran cannot be combined with Liencres, since the functionality of Chandrasekaran can't be

added to a cache memory, as cache memories do not operate as suggested by the Examiner.

Accordingly, Appellant submits none of the cited references taken either singly or in combination, teaches or suggests the combination of features recited in Appellant's claim 1.

Appellant's claims 14 and 26 recite features that are similar to the features recited in claim 1. Thus, for at least the above stated reasons, Appellant submits that the rejection of claims 1, 14, and 26 is in error and requests reversal of the rejection. The rejection of claims 2-13 (dependent from claim 1), claims 15-25 (dependent from claim 14), and claims 27-38 (dependent from claim 26) is similarly in error for at least the above stated reasons, and reversal of the rejection is requested. Each of claims 2-13, 15-25, and 27-38 recite additional combinations of features not taught or suggested in the cited art.

Separately argued dependent claims (by number)

Claims 3, 16 and 28 depend from claims 1, 14 and 26, respectively. Accordingly, the rejection of claims 3, 16 and 28 is in error for at least the reasons highlighted above with regard to claims 1, 14 and 26. Additionally, each of claims 3, 16 and 28 recite a combination of features including: "wherein the address packet is a read-to-own packet, the access right is a write access right, and wherein the system memory is configured to send the report corresponding to the read-to-own packet to the interface if a global access state of the coherency unit in the node is any global access state other than a modified global access state."

The Examiner asserts the above is taught at col. 7 of Liencres in "Read Transactions." Appellant disagrees. Liencres actually discloses

> When a memory request by the processor 21 cannot by fulfilled by the data in the processor cache memory 37, the processor cache controller 35 sends **a read request packet** across the cache bus 33 to the bus cache controller 31. The bus cache controller 31 proceeds to broadcast a corresponding read request packet across the memory bus 25. The read

> transaction initiated by the bus cache controller 31 consists of two packets: a read request packet sent by the bus cache controller 31 on the memory bus 25 and a read reply packet sent by another device on the memory bus. The read request packet contains the address of the memory requested by the processor cache controller 35 and is broadcast to all entities on the memory bus 25. A device on the memory bus 25 that contains the requested memory address responds to the read request packet with a read reply packet containing the subblock which includes the requested memory address. The read reply packet is generally issued by the main memory 23 except when the desired memory address is "owned" by another processor subsystem 20. In that case, the processor subsystem that owns the subblock must generate a read reply packet with the requested data.

From the foregoing, Appellant submits Liencres is merely teaching that in response to read request that can't be fulfilled by the cache memory 37, the cache controller 35 issues a read request packet to the bus cache controller 31, which may then broadcast a read request. Appellant fails to see how this teaches the limitations recited in Appellant's claim 3.

Appellant submits claims 16 and 28 recite similar features. Thus, for at least the above stated reasons, Appellant submits that the rejection of claims 3, 16 and 28 is in error and requests reversal of the rejection.

CONCLUSION

For the foregoing reasons, it is submitted that the Examiner's rejection of claims 1-38 is erroneous, and reversal of his decision is respectfully requested.

The Commissioner is authorized to charge any fees that may be due to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5681-01601/SJC.

Respectfully submitted,

/ Stephen J. Curran /

Stephen J. Curran
Reg. No. 50,664
AGENT FOR APPELLANT(S)

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8800

Date: May 27, 2008

VIII. APPENDIX

The claims on appeal are as follows.

1. A system, comprising:

a node including an active device, an interface to an inter-node network, a system memory, and an address network and a data network that is separate from the address network, coupling the active device, the interface, and the system memory;

an additional node coupled to the node by the inter-node network;

wherein in response to receiving from the active device an address packet initiating a transaction to gain an access right to a coherency unit, the system memory is configured to send a report corresponding to the address packet to the interface if the transaction cannot be satisfied within the node;

wherein the interface is configured to ignore the address packet and to send a coherency message requesting the access right to the additional node via the inter-node network in response to the report.

2. The system of claim 1, wherein the node includes a data network coupling the active device, the interface, and the system memory, and wherein the system memory is configured to send the report to the interface in a data packet.

3. The system of claim 1, wherein the address packet is a read-to-own packet, the access right is a write access right, and wherein the system memory is configured to send the report corresponding to the read-to-own packet to the interface if a global access state of the coherency unit in the node is any global access state other than a modified global access state.

4. The system of claim 3, wherein the node includes an additional active device, wherein the additional active device is configured to transition a read access right to the coherency unit to an invalid access right upon receipt of the read-to-own packet.

5. The system of claim 3, wherein the address network is configured to convey the read-to-own packet in broadcast mode, wherein the active device is configured to gain an ownership responsibility for the coherency unit upon receipt of the read-to-own packet.

6. The system of claim 5, wherein an additional interface included in the additional node is configured to receive the coherency message on the inter-node network, wherein the additional interface is configured to send a proxy address packet on an address network included in the additional node in response to the coherency message.

7. The system of claim 3, wherein in response to sending the coherency message, the interface is configured to receive an additional coherency message on the inter-node network;

wherein in response to the additional coherency message, the interface is configured to send data corresponding to the coherency unit to the active device.

8. The system of claim 7, wherein the active device is configured to gain the write access right to the coherency unit upon receipt of the data.

9. The system of claim 7, wherein the interface is further configured to send data corresponding to the coherency unit to the system memory in response to the additional coherency message, wherein in response to the data, the system memory is configured to update the global access state of the coherency unit in the node to the modified global access state.

10. The system of claim 3, wherein the system memory is configured to send data corresponding to the coherency unit to the active device if the global access state is the modified state and if the system memory has an ownership responsibility for the coherency unit, wherein the active device is configured to gain the write access right upon receipt of the data.

11. The system of claim 1, wherein the address packet is a read-to-share packet, the access right is a read access right, and wherein the system memory is configured to send the report corresponding to the read-to-share packet to the interface if a global access

state of the coherency unit in the node is not a modified global access state or a shared global access state.

12. The system of claim 1, wherein the interface is configured to add a record corresponding to the report to an outstanding transaction queue in response to receiving the report;

wherein the interface is configured to add a record to the outstanding transaction queue in response to each address packet specifying a coherency unit that is not mapped by the system memory.

13. The system of claim 12, wherein the interface is configured to send a corresponding coherency message on the inter-node network in response to each record in the outstanding transaction queue.

14. A node for use in a multi-node system, the node comprising:

a plurality of devices including a system memory, an active device, and an interface configured to send and receive coherency messages on an inter-node network coupling nodes in the multi-node computer system;

an address network configured to convey address packets between the plurality of devices;

a data network that is separate from the address network and configured to convey data packets between the plurality of devices;

wherein in response to receiving from the active device an address packet initiating a transaction to gain an access right to a coherency unit, the system memory is configured to send a report corresponding to the address packet to the interface if the transaction cannot be satisfied within the node;

wherein the interface is configured to ignore the address packet and to send a coherency message requesting the access right on the inter-node network in response to the report.

15. The node of claim 14, wherein the node includes a data network coupling the active device, the interface, and the system memory, and wherein the system memory is configured to send the report to the interface in a data packet.

16. The node of claim 14, wherein the address packet is a read-to-own packet, the access right is a write access right, and wherein the system memory is configured to send the report corresponding to the read-to-own packet to the interface if a global access state of the coherency unit in the node is any global access state other than a modified global access state.

17. The node of claim 16, wherein the node includes an additional active device, wherein the additional active device is configured to transition a read access right to the coherency unit to an invalid access right upon receipt of the read-to-own packet.

18. The node of claim 16, wherein the address network is configured to convey the read-to-own packet in broadcast mode, wherein the active device is configured to gain an ownership responsibility for the coherency unit upon receipt of the read-to-own packet.

19. The node of claim 16, wherein in response to sending the coherency message, the interface is configured to receive an additional coherency message on the inter-node network;

wherein in response to the additional coherency message, the interface is configured to send data corresponding to the coherency unit to the active device.

20. The node of claim 19, wherein the active device is configured to gain the write access right to the coherency unit upon receipt of the data.

21. The node of claim 19, wherein the interface is further configured to send data corresponding to the coherency unit to the system memory in response to the additional coherency message, wherein in response to the data, the system memory is configured to update the global access state of the coherency unit in the node to the modified global access state.

22. The node of claim 16, wherein the system memory is configured to send data corresponding to the coherency unit to the active device if the global access state is the modified state and if the system memory has an ownership responsibility for the

coherency unit, wherein the active device is configured to gain the write access right upon receipt of the data.

23. The node of claim 14, wherein the address packet is a read-to-share packet, the access right is a read access right, and wherein the system memory is configured to send the report corresponding to the read-to-share packet to the interface if a global access state of the coherency unit in the node is not a modified global access state or a shared global access state.

24. The node of claim 14, wherein the interface is configured to add a record corresponding to the report to an outstanding transaction queue in response to receiving the report;

wherein the interface is configured to add a record to the outstanding transaction queue in response to each address packet specifying a coherency unit that is not mapped by the system memory.

25. The node of claim 24, wherein the interface is configured to send a corresponding coherency message on the inter-node network in response to each record in the outstanding transaction queue.

26. A method of operating a multi-node system, wherein the multi-node system includes a node and an additional node coupled by an inter-node network, the method comprising:

an active device in the node initiating a transaction to gain an access right to a coherency unit by sending an address packet on an address network within the node;

an interface in the node ignoring the address packet;

in response to said sending the address packet, a system memory in the node sending, via a data network that is separate from the address network using a data packet, a report corresponding to the address packet to the interface if the transaction cannot be satisfied within the node; and

in response to the report, the interface sending a coherency message to an additional interface in the additional node via the inter-node network, wherein the coherency message requests the access right to the coherency unit.

27. The method of claim 26, wherein the node includes a data network coupling the active device, the interface, and the system memory, and wherein said sending the report comprises the system memory sending a data packet containing the report to the interface via the data network.

28. The method of claim 26, wherein the address packet is a read-to-own packet, the access right is a write access right, and wherein the method further comprises the system memory determining that the transaction cannot be satisfied within the node if a global access state of the coherency unit in the node is any global access state other than a modified global access state.

29. The method of claim 28, wherein the node includes an additional active device, wherein the method comprises the additional active device transitioning a read access right to the coherency unit to an invalid access right upon receipt of the read-to-own packet.

30. The method of claim 29, further comprising:

the address network conveying the read-to-own packet in broadcast mode; and

the active device gaining an ownership responsibility for the coherency unit upon receipt of the read-to-own packet.

31. The method of claim 30, further comprising:

an additional interface included in the additional node receiving the coherency message on the inter-node network; and

the additional interface sending a proxy address packet on an additional address network included in the additional node in response to the coherency message.

32. The method of claim 28, further comprising:

the interface receiving an additional coherency message on the inter-node network, wherein the additional coherency message is responsive to the coherency message;

in response to the additional coherency message, the interface sending data corresponding to the coherency unit to the active device.

33. The method of claim 32, further comprising the active device gaining the write access right to the coherency unit upon receipt of the data.

34. The method of claim 32, further comprising:

the interface sending data corresponding to the coherency unit to the system memory in response to the additional coherency message; and

in response to the data, the system memory updating the global access state of the coherency unit in the node to the modified global access state.

35. The method of claim 28, further comprising:

the system memory sending data corresponding to the coherency unit to the active device if the global access state is the modified state and if the system memory has an ownership responsibility for the coherency unit; and

the active device gaining the write access right upon receipt of the data.

36. The method of claim 26, wherein the address packet is a read-to-share packet, the access right is a read access right, and wherein said sending the report occurs if a global access state of the coherency unit in the node is not a modified global access state or a shared global access state.

37. The method of claim 26, further comprising:

the interface adding a record corresponding to the report to an outstanding transaction queue in response to receiving the report; and

the interface adding a record to the outstanding transaction queue in response to each address packet specifying a coherency unit that is not mapped by the system memory.

38. The method of claim 37, further comprising the interface sending a corresponding coherency message on the inter-node network in response to each record in the outstanding transaction queue.

IX. EVIDENCE APPENDIX

No evidence submitted under 37 C.F.R. §§ 1.130, 1.131, or 1.132 or otherwise entered by the Examiner is relied upon in this appeal.

X. RELATED PROCEEDINGS APPENDIX

There are no related proceedings.